TOPOLOGY VIA LOGIC

STEVEN VICKERS

Department of Computing, Imperial College of Science and Technology,
University of London

CAMBRIDGE
UNIVERSITY PRESS

Published by the Press Syndicate of the University of Cambridge
The Pitt Building, Trumpington Street, Cambridge CB2 1RP
40 West 20th Street, New York, NY 10011–4211, USA
10 Stamford Road, Oakleigh, Melbourne 3166, Australia

First published 1989
Reprinted 1990
First paperback edition 1996

British Library Cataloguing in Publication Data available

Library of Congress Cataloguing in Publication Data available

ISBN 0 521 36062 5 hardback
ISBN 0 521 57651 2 paperback

Transferred to digital printing 2003

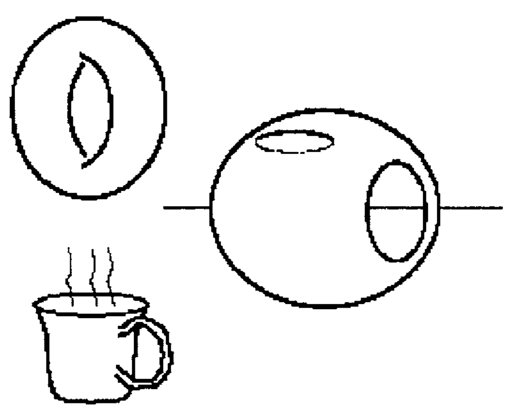
CONTENTS

INTRODUCTION

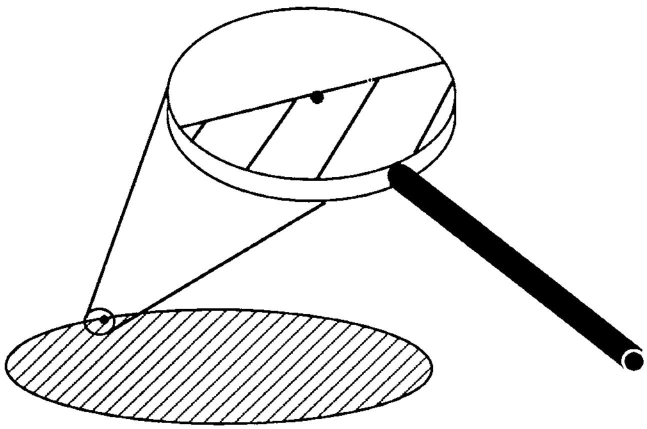
A Historical Overview

The origins of topology are very different from the context we shall be working in, and it is probably as well to compare some different ideas of what it is.

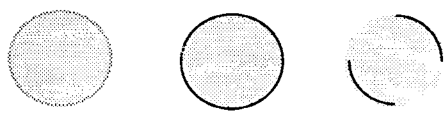
I – The first idea is that of *rubber sheet geometry,* that is to say geometry in which we don't mind stretching our space. This geometry is not at all concerned with distances or angles; it wants to answer questions like, "Is there a hole in this object?" (Although stretching is allowed, tearing isn't.) Martin Gardner [66] says, "Topologists have been called mathematicians who do not know the difference between a cup of coffee and a doughnut," the reason being that each has exactly one hole through it. According to rubber sheet geometry they are equivalent, because if they are made of stretchy enough material, one can be manipulated into the other. (The hole in the cup that counts is the one where you put your finger to hold it. The place where the coffee goes is a mere hollow. Of course, we are thinking of *ring* doughnuts.)



II – *The study of boundaries.* Since tearing is what makes a difference in rubber sheet topology, and since tearing creates new *boundaries* in the sheet, these would seem an important thing to look at. The characteristic of a boundary point of a set is that however closely you look at it, you can see some neighbouring points inside the set and some outside.



Part of this will entail studying *closed* sets, which include all their boundary points (like a circle with its circumference), and *open* sets, which include none of their boundary points (like a circle without its circumference). Of course, these are just extreme cases. There will also be sets that include some of their boundary points but not all.



open closed neither

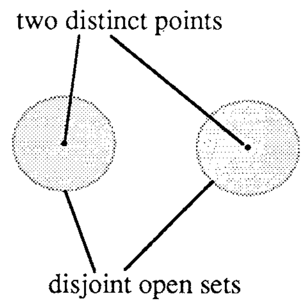
III – *The abstract study of open and closed sets.* The next step is one of abstraction. We forget all the geometry and just take an abstract set of "points", a *topological space*. We specify certain subsets as being *open* (their complements are the *closed* subsets), and we make sure that certain axioms, due to Hausdorff, are satisfied. Then we translate topological arguments from stage II into this abstract setting.

IV – *Locale theory.* The next step is to forget even about the points, and just take an abstract set of "open sets", with abstract algebraic operations to represent union and intersection. This structure is a *frame*. Sometimes, the points can be reconstructed from this frame of open sets.

This may seem like the ultimate in abstraction, but we shall see how considerations of *logic* make this an appropriate starting point from which to work backwards.

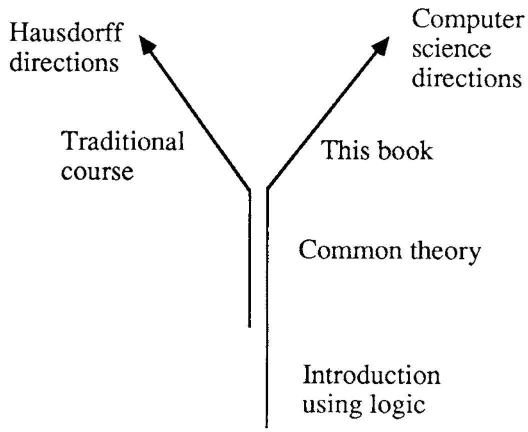
Hausdorff spaces

The topological axioms for open sets are very general and cover far more situations than just those arising from the rubber sheet ideas. Therefore in practice, topologists will apply extra axioms to restrict attention to the kind of space they're interested in. A very common one is the *Hausdorff separation axiom,* which says that any two distinct points can be "housed orf" from each other by disjoint open sets. The mainstream of topology deals with these *Hausdorff spaces.*



In computer science, however, topology is used to explain *approximate* states of information: the points include both approximate points and more refined points, and these relate to the topology by the property that if an open set contains an approximate point then it must also contain any refinement of it. Thus the approximate point and its refinement cannot possibly be "housed orf" by disjoint open sets and the topological space cannot possibly be Hausdorff. This means that topology as used in computer science – at least for the methods described here – runs in a different direction from the mainstream, even though it is still topology.

This book approaches topology in an unusual way, starting from frames and an explanation in terms of logic, and ends up with unusual applications – the non-Hausdorff topologies used in computer science. For computer scientists it is designed to provide a self-contained introduction, but as a route to the more traditional applications it is written to complement the standard introductions, of which there are many.



Other books to read

For rubber sheet geometry – browse through Martin Gardner's collections from his Scientific American column "Mathematical Puzzles and Diversions". Gardner [63] and [86] both contain relevant articles.

For a traditional approach to topology, giving greater emphasis to Hausdorff spaces, there is a wide choice of texts. A standard one is Kelley's "General Topology" [55].

For more on locale theory, an excellent book is Johnstone's "Stone Spaces" [82]. However, the later chapters do assume a good acquaintance with traditional topology.

Category theory

Interspersed throughout the text are remarks such as "categorically speaking, ...". These refer to category theory, and readers familiar with this will understand. The rest can ignore the remarks if they want. They indicate that we are, in a hidden way, using the methods of category theory. It is not necessary to know category theory to be able to understand this book, and in fact it is probably useful to see the methods in action informally before going on to the formal theory. The classic introduction is MacLane's "Categories for the Working Mathematician" [71], but a helpful one for computer scientists is the tutorial part of Pitt et al. [85].

Such remarks, and also those on other topics that are slightly off the main development, are often printed in a smaller typeface.